A Specific Decision Support System (SDSS) to Develop an Optimal Project Portfolio Mix Under Uncertainty

DENNIS S. KIRA, MARTIN I. KUSY, DAVID H. MURRAY, AND BARBARA J. GORANSON

Abstract—**A specific decision support system (SDSS) that can be used as a methodology for choosing an optimal portfolio mix of information systems projects is developed. An SDSS is developed by applying a number of well-known techniques in management science. A net present value (NPV) of the projects is maximized within limited resources and the solution of the model provides the optimal project start period, system development language, and staff size. The SDSS conducts a risk analysis on those variables that are deemed critical when determining the optimal solution. The final phase of the SDSS establishes the value of perfect information on these critical variables. To demonstrate the usefulness of the model, an example consisting of five information system (IS) projects are presented. It is shown that given the estimates of the exogenous parameters of the IS project environment, the model can determine when to begin the project development, which systems development language to use, and the number of systems development staff to assign to each project. Furthermore, the degree of the variability of the estimates of exogenous parameters is evaluated through empirical probability distributions.**

Keywords—**Specific decision support system (SDSS); information systems; optimal project mix; risk management; decision analysis; prior analysis; preposterior analysis; value of perfect information; uncertainty; projects.**

INTRODUCTION

IT HAS become devastatingly obvious to many firms that they can no longer conduct "business as usual" in order to remain competitive in the market. For this reason, corporations are becoming more acutely aware of the need for timely information. Information systems (IS's) which supply this information represent the single most important competitive edge the corporation has for its survival. Firms cannot afford to abdicate control of this area [27].

While rapid and continuous improvements in the underlying technology of information systems have encouraged the use of information as a competitive weapon, only a few advancements have been made in the development of management techniques in this area. In this paper, a specific decision support system (SDSS) is developed, using well-known management science techniques, to provide IS managers with an effective planning and decision-making tool. Here, we shall consider the SDSS to be the integration of hardware/software that allow a specific decision maker or group of them to deal with specific sets of related problems (see [35, pp. 10–11]). This SDSS provides them with an optimal mix of IS projects based on profit maximization criteria. Specifically, the system determines when to begin the project development, which systems development language to use, and the number of systems development staff to assign to each project. A typical use of SDSS in an uncertain environment is described, with a sample output of the analysis.

Modern corporations, faced with an increasingly volatile environment and fierce competition, often demand more from their IS development department than their existing staff resources can handle. The resulting backlog must be prioritized and scarce resources allocated to IS[1] projects in such a way that the choices maximize "value" for the corporation. These decisions will ultimately determine the nature and timing of the firm's information system, which is frequently a key factor in their ability to respond quickly to opportunities or competitive threats in the market place. A portfolio of new projects must be selected from the development backlog and resources assigned in such a manner that maximizes value under the constraints imposed by a limited staff. At present, IS planning lags behind most other corporate-level planning systems, despite the strategic importance of these systems for corporate survival and potential growth. Existing studies have found that effective IS planning systems are still relatively rare in contemporary computer centers [15], [19], [21].

The need for an effective framework in which a firm's IS project portfolio can be selected in an uncertain environment is readily apparent. Efficient use of IS resources can be achieved through the integration of systems, the minimization of system changes, and the effective prioritization and scheduling of competing IS projects. These projects should be evaluated in the same critical manner as any other major corporate investment such as plant equipment or the introduction of a new product line. In fact, IS planning can borrow from techniques used for capital budgeting allocation decisions, where scarce or limited resources and the presence of uncertainty are also major issues. In this paper, we propose a specific decision support system (SDSS) which utilizes a

Manuscript received June 22, 1988. This work was supported in part by NSERC Grant 06PIN020 and SSHRC Research Grant 410-87-0195.

D. S. Kira and M. I. Kusy are with the Department of Decision Sciences and Management Information Systems, Concordia University, Montreal, PQ H3G IM8, Canada.

D. H. Murray is with CN Rail, Montreal, PQ, Canada.

B. J. Goranson is with the Royal Bank of Canada, Montreal, PQ, Canada.

IEEE Log Number 8932182.

[1] IS projects consist of software developments, hardware decisions, and their implementation.

0018-9391/90/0800-0213$01.00 © 1990 IEEE

typical resource allocation model framework as a modeling component of the system [11], [35].

Finding the optimal mix of development projects over some defined planning interval is a formidable task even when number of critical variables and their relationships are known with certainty. Furthermore, the changing nature of IS systems introduces considerable uncertainty to many of the factors involved in the portfolio decision. These uncertainties can affect the estimation of project benefits, development time, and hardware requirements. There is an obvious need for a comprehensive set of tools and techniques to improve decision making in the complex and uncertain process of IS portfolio management. The proposed SDSS will provide a methodology that can be used to evaluate the risk associated with the estimates and their impact on profits explicitly. Information obtained, in turn, will assist the firm with its strategic and planning process. A detailed description of the proposed SDSS will be given in Section III.

IS Project Planning Tools

The most common forum for IS portfolio management decision making is the IS Steering Committee, according to the survey of 127 companies performed by Nolan [25]. A distinct majority (85 percent) of those companies claimed to use this approach to select projects and allocate staff, although most of the committees (73 percent) met no more frequently than every three months for a few hours. We can therefore assume that the formulation of the proposed IS portfolio is relegated to lower-level committees and the IS executive support staff. In these lower-level forums, decision making by consensus is typical, with only limited recourse to analytical tools [25]. There is very little explicit evaluation, therefore, of the tradeoffs between the risk and return of various projects.

A number of models have been suggested to improve this process. Among them is one suggested by Buss [3], in which the IS executive must form a composite ranking of potential projects based on financial costs and benefits, intangible benefits, technical importance, and conformity to the organization's objectives. Presumably by allowing intangible, technical, and strategic issues to enter into this ranking process, the uncertainty associated with profitability estimates can be avoided. Unfortunately, accountability for financial results and the ability to compare the financial benefits of different IS projects are lost in the process.

McFarlan [21] takes an important step by including risk assessment in his model. He argues that the size and structure of a project and the firm's experience with the technology involved are the chief determinants of risk. He does not directly determine the relationship between risk and return, however, and the methodology relies on a number of somewhat arbitrary surrogates to link expected return with degree of risk. Kanter [15], [16] uses a rating system to prioritize projects. This also depends on an arbitrary, intuitive scale. Another approach, presented by Long [19], uses a series of evaluators to form a consensus risk evaluation in a matrix format. Moore [22] and Putnam [28] both used similar techniques on different, but related, topics. Research and development (R&D) portfolio selection models utilizing myriad ranking procedures or mathematical programming methods are described in [5], [7], [10], [20], [33]. A risk index scheme, such as the one proposed by McFarlan requires the IS manager to know the total risk index of an IS applications portfolio, and have the ability to penalize "high-risk" projects. The optimal composition of IS projects is governed by the total risk index of the portfolio and the preassigned risk value. The proposed model does not assign the risk values to the individual projects but rather it considers the degree of uncertainty associated with the quantitative estimates of the environmental variables (for example, a large standard error). That is, for our purposes risk can be defined to be the variability of the parameters being estimated. Of course, this is related to the McFarlan's risk index measure since the difficulties in estimating the environmental variables leads to the project being classified as high risk [21]. While one can adopt risk-adjusted project portfolio selection procedures [6], we believe it would be perferable to develop a method which would allow the IS manager to view the uncertainty of estimates explicitly in terms of its impact on profits. The risk-return construct is portrayed by the probability distribution of NPV.

In this paper, an SDSS system based on both integer linear programming and Monte Carlo simulation techniques is described. It incorporates the following features, which we believe to be essential for an IS portfolio management model in today's competitive environment.

1) *Multiperiodicity:* Staff allotment, the time-value of money and the improvement of price performance of computer hardware can all be included as a function of time.

2) *Effect of Uncertainty:* Sensitivity analysis is performed to examine the possible effects on expected return, cost, and project size.

3) *Development Technology:* Tradeoffs concerning development staff productivity and hardware costs are considered.

4) *Diminishing Marginal Productivity:* The reduction in productivity of each additional person assigned to a project is also taken into account in this model.

5) *Value of Additional Information:* The information phase of the SDSS contains a technique for assessing the value of a further study to reduce uncertainty.

The SDSS Model

The DSS/SDSS described in this paper, defined here as a computer system designed to aid management decision making, Sprague and Carlson [29] and Vazsonyi [32], is conceptually divided into three distinct phases: deterministic, probabilistic, and informational. The second two phases use the output of the preceding phase as input. The initial, or deterministic phase, is generated using data determined by the IS manager according to his own environmental estimates and assessment of project characteristics. The SDSS software package is written in Statistical Analysis System (SAS), a general-purpose data analysis language with extensions for a mathematical programming, graphic, and report-writing facilities. The SDSS software runs on IBM 370/30XX/43XX and other mainframes. It is possible to adopt this code for the micro environment with 80386 processor and we hope to accomplish this in near future. Each phase will now be examined.

Deterministic Phase

The various inputs required for the integer linear program used during this phase can be categorized as follows: a performance measure (for the objective function), exogenous parameters, policies and constraints, decision variables, and intermediate variables. These inputs are generated by using data determined by the IS manager according to his own environmental estimates and assessment of project characteristics. Although, the estimations of project characteristics such as size, complexity, effort, and benefit are a difficult task for IS managers there are some well-known estimation models he/she can utilized. For example, the COCOMO model estimates the man-month development effort as well as the software development time [2]. SLIM [28] and ESTIMACS [2] are software packages that can provide cost estimates of the projects. However, Boehm states that the estimation of these project parameters is a very difficult task and concludes that the estimates obtained are often very different from the actual realizations. Empirical findings of the reliability of the cost estimation models are reported by Kemerer [17] where he concludes that all of the models tested failed to sufficiently reflect the underlying factors affecting productivity. In view of these estimation difficulties, the major contribution of the proposed model, we believe, is its ability to assess explicitly the value of additional information in estimating number of key variables affecting both the profit level and the probability distribution of profits.

The chosen *performance measure* consists of the total net benefit of all projects completed within the planning horizon, expressed in present value terms. This value is the sum of the differences between the value of actual benefits and the actual dollar investment in hardware for all projects selected. Although number of measurements can be used as a performance measure of the project, the net present value (NPV) criterion is the only one that is necessarily consistent with the maximization of shareholder's wealth [6]. For this model, we shall assume usual economic theory where the firm is assumed to maximize shareholder's wealth.

Exogenous parameters, estimated by the IS manager, include the following:

1) Net project benefits, excluding hardware and systems development costs but including all other costs, are expressed in terms of equivalent monthly savings in current dollars which would result from an immediate implementation of the project. We note that these benefits will begin to flow only when the system's development is complete, and so it is assumed that these benefits will be reduced by the time-value of money associated with the delays in 1) beginning and 2) conducting the systems development. Therefore, these values are discounted by firm's weighted average cost of capital (WACC) [6].

2) The Price-performance improvement factor (PPI), reflecting the declining cost per unit of computing power and the lower price of computer hardware with time, must be predicted for the planning horizon in question.

3) Program language efficiency is expressed in terms of a "language efficiency multiplier" for each type of language, given that high-level languages require considerably more computing hardware than lower level languages. For example, if conventional languages such as COBOL are given a multiplier of 1, and fourth generation languages such as FOCUS are given a multiplier of 10. The estimates of these values are partially based on work due to Jones [14] and one of the author's experience in the MIS department at Canadian National Rail. Of course, these estimates can be adjusted for the particular organization and its environments. Here, we are also assuming that the development activities are affected by the choice of programming language.

4) The cost of additional computer hardware required for each project must be expressed in current dollars (as if purchased immediately), assuming that a conventional programming language has been used (normalized, see Jones [14]).

5) Development effort "size" is frequently measured in "function points" [14], [18], which are based on a number of factors related to size and complexity, i.e., number of user inputs, user outputs, inquiries, logical master files, and logical interfaces to other systems. Function points are derived using an empirical relationship based on countable measures of the software's information domain (such as the number of user inputs and outputs) and subjective assessments of software complexity. Albrecht and Gaffiney [1], based on their empirical study, used the following weights for the complexity measure in calculating the function point: for the input (4), for the output (5), for the number of inquiries (4), for the number of master files (10) and, for the number of interfaces (7). The final sum of the products of the functions and their weights is called a function point. This methodology can be used to estimate the number of lines of source code for any programming language (see Jones [14, p. 77]).

6) Labor productivity estimates are based on two known factors related to project labor efficiency: the greater programming efficiency of higher level languages versus lower level languages and the diminishing marginal productivity rate associated with each additional worker added to a project for each language type.

Policies and constraints, provided by the IS manager, are used as input parameters during this phase. Three such inputs are included in this model:

1) A fixed programming staff during the planning horizon is assumed. This model provides a short-run view in the sense that, during the planning period, the size of the programming staff cannot be increased due to the chronic shortage of qualified people in the labor market, and will not be reduced due to the long-run importance of retaining such people for future undertakings. The cost of retaining this staff is therefore a sunk cost in this analysis.

2) Prior to input, net present values must be established. Specifically, the firm's weighted average cost of capital (WACC) is used in this study as a discount rate.

3) A fixed planning horizon, which may consist of several planning periods, must be established by the IS manager.

Intermediate variables must be determined for each project. For the purpose of this exercise, three key variables are included:

1) The duration of development effort for each project, a

function of language selected and staff assigned to the project, must be established.

2) Actual benefits for each project must be determined as a function of stated project benefits, starting time of the development, the duration of the development effort, and the discount rate employed.

3) Hardware costs required to support the implementation of each project are based on current hardware costs, the price-performance factor, a language efficiency multiplier, start and finish time for the development effort, and the discount rate.

Finally, the *decision variables* produced by the integer linear program will provide the IS manager with the following four optimal decisions, based on the inputs outlined above:

1) Which projects to develop.

2) When to begin the development of each chosen project.

3) The number of people to assign to each project.

4) The appropriate programming language for a given project.

A net benefit is computed for each combination of decision variables, and an integer linear program is constructed with these net benefits as objective function coefficients to maximize total net benefits subject to fixed staff constraints within a given horizon.

The relationships discussed previously provide the foundation for an integer linear programming model which enables the IS manager to make an optimal choice of projects, starting times, programming languages, and staff levels. Each of the projects being considered can be thought of as a number of mutually exclusive subprojects, one for each combination of programming language, staff assignment size, and starting time. A net benefit can be computed for each such combination and an integer linear program (ILP) then constructed with these net benefits as objective function coefficients. This ILP is formulated to maximize total net benefits subject to fixed staff constraints within a given horizon. A detailed description of the modelling component of the SDSS is as follows:

Let

- P be the number of projects under consideration,
- L be the number of possible types of programming language,
- S be the number of possible staff allotments. That is, there are S such allotments, each with a value of, say N_s, $s = 1, 2, 3, \cdots, S$,
- Q be the number of periods in the planning cycle,
- T be the total number of staff available;

$$x_{plsq} = \begin{cases} 1 & \text{if project } p \text{ begins development using} \\ & \text{language type } l \text{ and staff allotment } s \\ & \text{in period } q, \text{ for } 1 \le p \le P, l \in \{1, 2, \cdots, L\}, \\ & 1 \le s \le S, \text{ and } 1 \le q \le Q \\ 0 & \text{otherwise;} \end{cases}$$

$$z_{plsq} = \begin{cases} 1 & \text{if the development of project } p \text{ using} \\ & \text{language type 1 with staff allotment } s \\ & \text{is active during period } q, \text{ for all } p, l, s, q \\ 0 & \text{otherwise;} \end{cases}$$

- NPV_{plsq} be the net benefit from subproject $plsq$, as computed in the model described earlier (this essentially is the value added to the firm by adopting the project at period p with the appropriate resource composition),
- DUR_{pls} be the number of periods required to develop project p using language l with staff allotment s, i.e., it is the project duration of the project commencing at period p.

We wish to maximize:

$$\sum_{plsq} \mathrm{NPV}_{plsq} x_{plsq}$$

subject to:

1) staffing constraints in each period:

$$\sum_{pls} N_s z_{pls1} \le T$$

$$\vdots$$

$$\sum_{pls} N_s z_{plsQ} \le T$$

2) mutual exclusivity constraints:

$$\sum_{lsq} x_{1lsq} \le 1$$

$$\vdots$$

$$\sum_{lsq} x_{Plsq} \le 1$$

3) period-linking constraints:

$$-x_{pls1} + z_{pls1} \le 0$$

$$-x_{pls2} - z_{pls1} + z_{pls2} \le 0$$

$$\vdots$$

$$-x_{plsQ} - z_{pls(Q-1)} + z_{plsQ} \le 0, \quad \text{for all } p, l, s$$

$$\sum_q z_{plsq} = \mathrm{DUR}_{pls} \sum_q x_{plsq}$$

4) integer constraints:

$$x_{plsq}, z_{plsq} = 0, 1, \qquad \text{for all } p, l, s, q.$$

For larger values of P, L, S, and Q, the problem becomes quite large. It contains $2^*P^*L^*S^*Q$ integer variables and $P + Q + (Q+1)^*P^*L^*S$ constraints. If, for example, there are 10 projects under consideration ($P = 10$), three possible levels of

programming language ($L = 3$), 100 possible staff allotments ($S = 100$) corresponding to a total staff of $T = 100$, and 60 months in the planning interval ($Q = 60$), the problem will involve 360 000 variables and 183 070 constraints.

Suggested simplifications which will not significantly reduce the usefulness of the results include:

1) Reducing the number of time periods by introducing quarter periods (i.e., 3 months at a time) as well as limiting the total period considered (i.e., from 5 to 3 years) alters the number of periods considered from 60 to 12.

2) Reducing the number of possible staff sizes to be allocated to each project in a manner which favors small sizes to reduce the effect of diminishing marginal productivity of large groups will result in fewer group sizes used. For example, instead of considering every combination possible, only 8 staff sizes could be considered.

3) Depending on the firm, it is possible that only 2 levels of language may be realistically included, rather than 3.

The resulting example would now have only 3840 variables and 2102 constraints. Further reductions could be achieved through the exclusion of subprojects which are found to have hardware costs which exceed their benefits or cannot be completed within the planning horizon established.

In summary, the deterministic phase of the SDSS generates an integer linear program customized by the specific values of the exogenous and policy parameters of the model user. An IP solution is then generated to obtain that combination of projects, start times, and language levels which optimizes the net present value of the project mix. This optimal IS project composition should provide the firm with "maximum" or "best" use of their information systems resources since it is determined within the scope of firm's strategic planning (e.g., the calculation of benefits, the policy constraints).

The next phase of the SDSS addresses the important issue of uncertainty. Zmud [42] states that "An important insight to understanding the problems associated with managing software development is that most difficulties can be traced to the uncertainty that pervades software development."

Probabilistic Phase

While risk analysis has been used profitably to identify high-risk projects upon which to focus management attention during development, its role in IS project selection normally involves the assignment of a risk index of some type and the manager is expected to combine IS projects so that the total "portfolio" risk index is within the given range.

While such schemes could easily be incorporated into the deterministic phase of this model, it would be preferable to develop a method which would allow the IS manager to view an estimation risk explicitly in terms of its impact on profits.

We will now introduce uncertainty to the modeling component of the SDSS, and observe how the profitability of the optimal decision found by the integer linear program in the deterministic phase is affected. In this manner, the risk associated with the quantitative estimates of the exogenous parameters can be considered explicitly in terms of its impact on profits. This is accomplished by allowing some of the exogenous parameters, which so far have been considered fixed, to become random variables. Values for these will be generated using a Monte Carlo simulation. Although it is much more realistic to represent these exogenous parameters by dependent random variables, the computational burden, especially for the informational phase of the model makes it somewhat untractable. In any event, we believe that the procedure suggested here reflects the actual situation sufficiently well and remains computationally tractable for realistic problems. The pioneering work of utilizing the Monte Carlo method in project selection problems is largely due to Hertz [13]. Spetzler [34] provides a similar method of dealing with risk where a corporate capital investment decision must be made. Three parameters have been chosen to become stochastic in our example on the basis that they are most frequently the source of serious estimation errors in IS planning: project benefits, hardware costs and project size. We assume that their probability distributions, which may be developed from historical data or subjective assessment, are known. Realizations are then generated according to their respective distributions. A new optimal solution and its benefits are computed for each of the realizations generated for these parameters, which we had assumed to be fixed in the deterministic phase. When this process is repeated many times, a cumulative probability distribution function of the total net benefits under these conditions of uncertainty can be developed.

Certain feasible situations must be dealt with when trying out each of these new possibilities. In the case where a staff shortfall is introduced through the changing variables, the simulation assumes that temporary programmers will be obtained from an outside source at double the gross salary for the period in question. It is also assumed that a temporary group will be employed to complete any work beyond the planning horizon, under the same salary conditions. We note that this situation does not occur in the deterministic phase since the exogenous parameters are constant.

The SDSS allows the random variables to be considered individually as well as in combination. The final output for each selection of variables to be analyzed is a cumulative probability distribution of total net benefits which can be used to assess the sensitivity of this value to random changes in the selected parameter(s).

Informational Phase

Having made a posterior analysis to determine the degree of variability of profitability of the optimal solution under uncertainty, the IS manager may wish to reduce this uncertainty in specific parameters if the variability is at unacceptable levels. Various measures, such as work studies to establish project benefits more firmly, or a more detailed system design to establish project size and hardware requirements, can be performed for certain projects at this stage. However, the delays associated with further studies and the costs incurred by these studies may exceed potential benefits. An estimate of the value of this extra information would be beneficial before commissioning extra design work or studies. The procedure by which this value is obtained is generally referred to as preposterior analysis.

In order to establish this value, this is, the value of per-

TABLE I

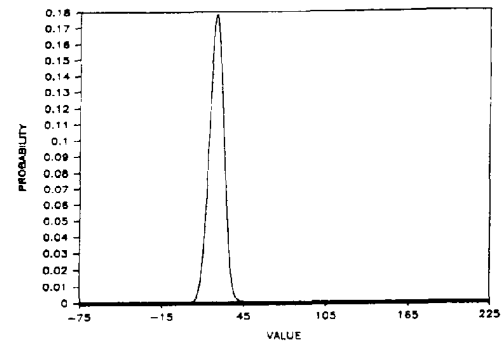
PROJECT NUMBER	EXPECTED BENEFITS (/month)	STANDARD DEVIATION	EXPECTED INVESTMENT (Hardware)	FUNCTION POINTS
1	$ 25,000	$ 5,000	$ 500,000	3,800
2	$ 35,000	$ 5,000	$ 500,000	7,000
3	$ 40,000	$ 4,000	$ 60,000	7,600
4	$ 75,000	$ 30,000	$ 900,000	7,500
5	$ 100,000	$ 50,000	$ 200,000	9,500

fect information, the integer linear programming of the first phase and the Monte Carlo simulation techniques of the second phase must be combined. The original optimal solution of the deterministic phase is evaluated under each of the random "realizations" generated during the probabilistic phase, using the same assumptions concerning infeasibility as were used earlier (temporary staffing). These benefits will then be compared with those of the new solutions generated under the same realizations during the probabilistic phase. The new optimal solution, by definition, will yield higher benefits than the deterministic solution which was generated under different conditions. The difference represents the increase in total benefits that would have been obtained if the realizations generated on a random basis had been forecast, and the new optimal solution had been implemented instead of the original solution generated during the deterministic phase. This process is repeated until a cumulative frequency distribution emerges for all of these differences. This cumulative distribution function represents the value of perfect information for the chosen variables. It provides the relative frequency that a minimum amount of money can be saved if the IS manager can foresee the future and make an optimal decision accordingly.

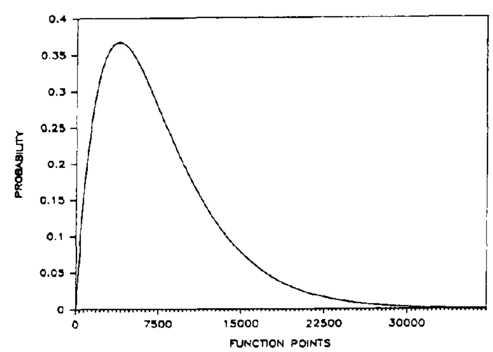
SDSS Results

It is not within the scope of this paper to detail all the results that can be obtained by the SDSS model. We will use an example which illustrates those analyses deemed of general interest and the type of results which are considered most useful. In this example, we assume that the IS manager of a medium-sized installation has a total programming staff of 50 people. Five possible projects were identified, in conjunction with the user departments of the CN Rail by one of the authors. Initial feasibility studies and external designs have been conducted, resulting in the following information in Table I.

The "standard deviation" figures in Table I refer to the project benefits. It has been determined through discussions with users that the project benefits are normally distributed. The mean and standard deviation estimates are use to generate the appropriate distributions. The distributions of project size and hardware investment, on the other hand, have been found to be Erlang variates. While these distributions are consistent with our experience, the SDSS provides the ability to generate random variates from a large number of other distributions. The IS manager thus can select from among these (e.g., Beta distribution) and incorporate them into the model. The shapes of the benefits and project size distributions of project 1 are illustrated in Fig. 1. In this example, we have



(a)



(b)

Fig. 1. Probability distributions of (a) project 1 benefits distribution and (b) project 1 size distribution.

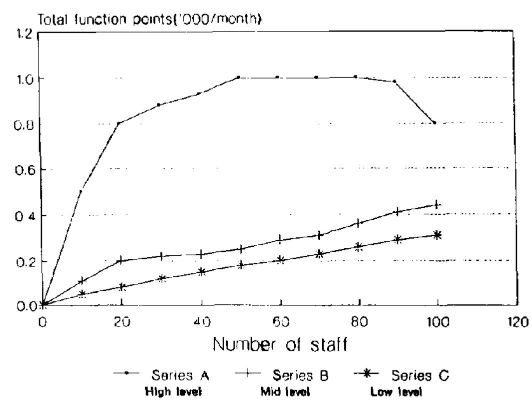


Fig. 2. Labor productivity curves by language type, staff size.

set $T = 50$, $Q = 12$, $P = 5$, $L = 2$, $S = 8$, WACC = 0.16, and PPI = 0.1. The hardware efficiency factors for lower level and upper level are 1 and 10, respectively, and the labor productivity rates are as indicated in Fig. 2. An example calculation of net present value using one possible set of decision variables is in Appendix I. The deterministic phase of this problem required 5542 iterations, and consumed 114 minutes of computer time on an IBM-compatible processor with an execution speed of approximately 6.5-million instructions per second. SDSS initially creates an input file with project

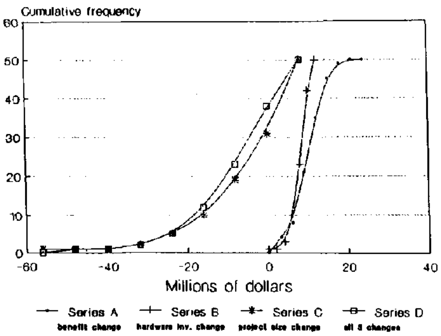


Fig. 3. Sensitivity analysis: parameter changes.

information, and constructs and then executes the integer program to select an optimal solution. The optimal solution is as follows: begin project 3 immediately, use an upper level language, and assign 19 people. The project will require 9 months to complete. Begin project 5 immediately, also using an upper level language, and assign 31 people. It will be completed in 9 months. When projects 3 and 5 are complete, assign all 50 people to project 4 using a lower level development language. This project will then require two years to complete. Following this plan will result in net benefits with a present value of $11 300 300.

The probabilistic phase of SDSS was performed allowing the following parameters to vary: benefits, hardware investment and size estimates. Each one was allowed to vary on an individual basis and then all three exogenous variables were made to vary simultaneously. In each case, 500 iterations of the Monte Carlo simulation process were performed. The resulting cumulative distribution functions are reproduced in Fig. 3. We note that the project size estimate appears to be the most critical parameter among the 3 parameters we chose, since it has the highest variation. We also note the relative insensitivity of benefit and hardware parameters.

The final (informational) phase of the SDSS can provide some assistance in determining the amount of effort the IS manager should spend in obtaining better estimates of these parameters. The SDSS was run allowing benefits, hardware investment, and project size to vary simultaneously.

In total, 100 iterations of the Monte Carlo simulation in the informational phase were performed, each involving the construction and solution of a large integer linear program. Computer time of 1398 minutes on a 6.5-million- instructions-per-second processor were required to conduct this analysis. Here, the IS manager needs to determine the "value" of additional information obtained since the informational phase of the SDSS requires considerable amount of computer time. It is recommended that this part of the model should employed for the cases where good estimates of the exogenous variables are not available to the IS manager.

The cumulative frequency distribution of *differences* between the net present value of the optimal solution for each random state of nature and those of the original deterministic phase solution evaluated under the conditions represented by the random state of nature is presented in Fig. 4.

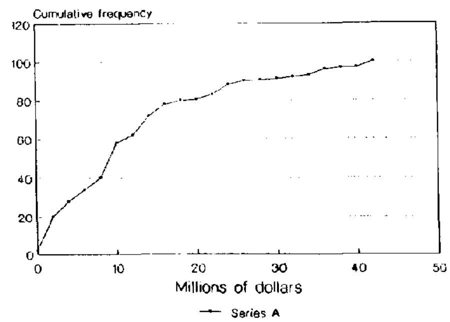


Fig. 4. Value of perfect information: all variables.

On the basis of this information, the IS manager would be justified in spending up to $2 million on activities that provide better estimates of the three parameters allowed to vary in this example. This follows from the cumulative graph of Fig. 4, since there is a very good chance (greater than 90 percent) that EVPI exceeds $2 million. Here, of course we should note that an additional information does not always add "value" to the decision making process. This follows since no amount of additional information can alter the "optimal" decision. In such a case, the information is valueless and any cost incurred in gathering additional information is a sunk cost. The numerical value of additional information, the expected value of sample information (EVSI), is the difference between the expected value of the optimal decision using the sample information and the expected value of the optimal decision without any additional information (EV). This value range from zero to an upper bound of the expected value of perfect information (EVPI), depending upon the reliability of the estimates.

The principles of decision analysis can be applied to develop an effective SDSS for the IS project planning process. Unlike previous models, the SDSS model in this paper explicitly incorporates uncertainty and examines the impact of uncertainty of important factors in IS planning on the firm's profitability.

Throughout our discussions, we have assumed that IS managers can provide appropriate subjective or objective assessments of exogenous parameters such as benefits, project size, and hardware costs according to his own environmental estimates and assessment of project characteristics. Although an estimation of project complexity, effort, and benefit are a difficult task for IS managers there are some estimation models he/she can utilize. For example, estimation models such as COCOMO, SLIM, and ESTIMACS can assist managers in developing quantitative estimates. If this is not feasible, then the subjective estimation procedures such as expert resolution (see Morris [23], [24]) can be utilized in estimating the parameters.

In this study, the risk is measured by the degree of uncertainty associated with the exogenous parameters such as project benefits, Price-performance improvement (PPI), "language efficiency multiplier" for each type of language, size, and complexity, and labor productivity level. An IS manager can utilize this information to establish the precise relationship between risk and expected return related to particular

uncertain variables which are encountered when implementing IS projects. By providing a tool by which the IS planner can evaluate the value of additional information, studies to reduce uncertainty of the estimates can be assessed in terms of their cost justification. After a final iteration of the process contained in this SDSS, an IS manager can provide the steering committee executives with firm recommendations for staff allocations, scheduling, and appropriate development languages for each project scheduled. He or she can also justify their choices based on a satisfactory risk-return relationship and the probable impact on profitability of alternative solutions.

In our example, three parameters have been chosen to become stochastic on the basis that they are most frequently the source of serious estimation errors in IS planning: project benefits, hardware costs, and project size. We have assumed that their probability distributions, which may be developed from historical data or subjective assessment, are known to the IS managers.

There are number of future improvements that can be made on SDSS to enhance the usefulness of this model. To begin with, one can develop an interactive estimation models of exogenous parameters such as language efficiency multiplier, function points, and labor productivity factor. These models can then be interfaced with the SDSS to better reflect the manager's ability to estimate these environmental factors. This could take the form, for example, of an interactive computer session between the "experts" and the SDSS which constructs the appropriate probability distributions for critical variables based on the answers to a series of pertinent questions. Second, the number of variables which are allowed to take on random values could also be increased. Examples are labor productivity, planning horizon, future cost of hardware, and language efficiency. Of course, the care must be exercised in expanding the number of factors considered as a stochastic factor since this will increase the computational burden considerably. Finally, the computational aspect of the model can be improved upon by utilizing more efficient language with the additional programming efforts. Also, one can develop a micro version of the model, perhaps by using the programming language such as C.

As a final note, the cost of developing a complete DSS as discussed above is in the order of 2 to 3 person years. Although initially this investment may seem somewhat large, if it is compared to the cost of supporting less than optimal sets of projects, then these developmental costs shrink in comparison.

Appendix I

A typical scenario that illustrates the use of various parameters is described in this appendix.

Suppose project A provides \$30 000 monthly benefits when implemented and the development effort has been estimated to be 6 000 function points. Moreover, if purchased today, the hardware to support the implementation of project A would cost \$100 000 if the project used an intermediate level language such as COBOL. It is further known that the hardware efficiency factors are estimated as: low level language (e.g., Assembler) = 0.5; intermediate level language (e.g., COBOL) = 1.0; high level language (e.g., Focus) = 10.0.

Also the IS manager estimates the price performance of hardware to fall by 10 percent annually during the next three years. That is, PPI = 0.1. The company's weighted average cost of capital (WACC) is 16 percent, and the Fig. 2 of the text expresses the productivity of labor.

Finally, supposed that the IS manager has decided to develop project A beginning one year from now. The project will be developed using FOCUS, and 10 programmers will be assigned to its development (a particular set of decision variables).

The problem is to determine the net benefit of this decision. The intermediate variables are computed as follows:

Project Duration: Ten programmers can produce 500 function points per period (month) if programming is done in a high level language (see Fig. 2). Therefore, the project will require one year to develop.

Actual Benefits: Development will begin in one year and will require one year to complete. The stated benefit of \$30000 per month is a perpetuity with a value of \$30000 divided by the *monthly* cost of capital, but must be discounted to account for the two-year delay. The present value of project A is therefore

$$(\$30000/((1.16)^{-1/12} - 1))^*((1.16)^{-2}) = \$1\,791\,454.$$

The stated hardware investment of \$100000 must be multiplied by a hardware efficiency factor of 10 to account for the high level programming language selected. It must then be multiplied by $(0.9)^*(0.9)$ to account for the fact that the purchase will only occur in two years, and that the price per unit of hardware is falling by 10 percent annually. Finally, it must be adjusted by the time-value of money. The appropriate calculation is as follows:

$$\$100000^*(0.9)^{2*}10^*((1.16)^{-2}) = \$601\,962.$$

The net benefit from project A under these conditions is equal to

$$\$1\,791\,454 - \$601\,962 = \$1\,189\,492.$$

This example illustrates how the value of a single project is determined, given a choice of development language, a starting time, and a staffing level. Similar computations would be performed for each project to be undertaken, and the total net benefit of this specific set of decision variables would simply be the sum of the individually computed net benefits.

Acknowledgment

The authors thank the referees for their comments.

References

[1] A. J. Albrecht and J. E. Gaffiney, "Software function, source lines of code, and development effort production: Software science validation," *IEEE Trans. Software Eng.*, vol. SE-9, no. 6, pp. 639–647, Nov. 1983.

[2] Boehm, B. W. *Software Engineering Economics*. Englewood Cliffs, NJ: Prentice-Hall, 1981.

[3] D. J. Buss, "How to rank computer projects," *Harvard Bus. Rev.*, pp. 118–125, Jan.–Feb. 1983.

[4] J. I. Cash, F. W. McFarlan, and J. L. McKenney, *Corporate Information Systems Management: Text and Cases*. Homewood, IL: Irwin, 1983, p. 552.

[5] T. E. Clark, "Decision making in technically based organizations: A literature survey of present practice," *IEEE Trans. Eng. Manag.* vol. EM- 21, pp. 9–23, Feb. 1974.

[6] T. E. Copeland, and J. F. Weston, *Financial Theory and Corporate Policy*. Reading, MA: Addison-Wesley, 1983, p. 795.

[7] B. V. Dean, "Evaluating selection, and controlling R&D projects," *AMA Res. Study*, vol. 89, 1968.

[8] D. E. Emrick, "Considering computer resource consumption in the selection of appropriate development software," *Electron. Data Process. Perform. Rev.*, vol. 13, no. 11, pp. 1–6, 1985.

[9] A. Figenbaum and H. Thomas, "Dynamic and risk measurement perspectives on Bowman's risk-return paradox for strategic management: An empirical study," *Strategic Manag. J.*, vol. 7, pp. 395–407, 1986.

[10] A. E. Gear, A. G. Lockett, and A. W. Pearson, "Analysis of some portfolio selection models for R&D," *IEEE Trans. Eng. Manag.*, vol. EM- 18, pp. 66–87, 1971.

[11] L. A. Gordon and G. E. Pinches, *Improving Capital Budgeting: A Decision Support System Approach*. Reading, MA: Addison-Wesley, 1984, p. 116.

[12] P. G. Hall and T. J. Lincoln, "A cost effectiveness study of production and engineering systems within a tightly integrated manufacturing environment," *Manag. Datamatics*, vol. 5, no. 6, pp. 261–273, 1976.

[13] D. B. Hertz, "Risk analysis in capital investment," *Harvard Bus. Rev.*, vol. 42, no. 2, pp. 95–106, 1964.

[14] C. Jones, *Programming Productivity*. New York: McGraw-Hill, 1986, p. 280.

[15] J. Kanter, *Management Information Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1984, p. 540.

[16] ——, "MIS long-range planning: Why don't more companies do it?," *Infosystems*, 1982.

[17] C. F. Kemerer, "An empirical validation of software cost estimation models," *Commun. Ass. Comput. Mach.*, vol. 30, no. 2, pp. 414–429, 1987.

[18] B. Konsynski and J. Kotternann, "Complexity measures in system development," in *Proc. Conf. Information Syst.* (Cambridge MA), 1981, pp. 173–199.

[19] E. L. Long, *Design and Strategy for Corporate Information Services*. Englewood Cliffs, NJ: Prentice-Hall, 1982.

[20] G. R. Madey and B. V. Dean, "Strategic planning for investment in R&D using decision analysis and mathematical programming," *IEEE Trans. Eng. Manag.*, vol. EM-32, no. 2, pp. 84–90, May 1985.

[21] F. W. McFarlan, "Portfolio approach to information systems," *Harvard Bus. Rev.*, pp. 142–150, Sept.–Oct., 1981.

[22] J. H. Moore, "A framework for MIS software development projects," *MIS Quart.*, vol. 3, Nov. 1, pp. 29–38, 1979.

[23] P. A. Morris, "Decision analysis expert use," *Manag. Sci.*, vol. 20, no. 9, pp. 1233–1241, 1974.

[24] ——, "Combining Bayesian approach," *Manag. Sci.*, vol. 23, no. 7, pp. 679–693, 1977.

[25] R. L. Nolan, "Managing information systems by committee," *Harvard Bus. Rev.*, pp. 72–79, July–Aug. 1982.

[26] R. L. Nolan and C. F. Gibson, "Managing the four stages of EDP growth," *Harvard Bus. Rev.*, pp. 76–85, Jan.–Feb. 1974.

[27] M. E. Porter and V. E. Millar, "How information gives you competitive advantage," *Harvard Bus. Rev.*, pp. 149–160, July–Aug. 1985.

[28] L. H. Putnam, "A general empirical solution to the macro software sizing and estimation problem," *IEEE Trans. Software Eng.*, vol. 4, no. 4, pp. 345–361, 1978.

[29] P. J. Pyburn, "Linking the MIS plan with corporate strategy, an exploratory study," *MIS Quart.*, 1983.

[30] D. G. Robinson, "Synchronizing systems with business values," *Datamation*, June 1984.

[31] W. E. Souder, "Achieving organizational consensus with respect to R&D project selection criteria," *Manag. Sci.*, vol. 21, no. 6, pp. 669–691, 1975.

[32] ——, "Comparative analysis of R&D investment models," *AIIE Trans.*, vol. 1, no. 2, pp. 57–64, 1972.

[33] ——, "Selecting and staffing R&D projects via operations research," *Chem. Eng. Progress*, vol. 63, no. 11, pp. 27–37, 1967.

[34] Spetzler, C. S., "The development of a corporate risk policy for capital investment decisions," *IEEE Trans. Syst., Man, Cybern.*, pp. 279–300, Sept. 1968.

[35] R. H. Sprague, Jr. and E. D. Carlson, *Building Effective Decision Support Systems*. Englewood Cliffs, NJ, Prentice-Hall, 1982, p. 329.

[36] R. D. Stewart, *Cost Estimating*. New York: Wiley, 1982.

[37] R. J. Thierauf, *Decision Support Systems for Effective Planning and Control. A Case Study Approach*. Englewood Cliffs, NJ: Prentice-Hall, 1982, p. 536.

[38] A. Vazsonyi, "Decision support systems: The new technology of decision making?" *Interfaces*, Nov. 1978.

[39] M. A. Wallach, N. Logan, and D. J. Ben, "Group influence on individual risk taking," *J. Abnormal Soc. Psychol.*, vol. 65, pp. 75–86, 1962.

[40] T. C. Wiloughby, "Origins of systems projects," *J. Syst. Manag.*, Oct. 1975.

[41] J. A. Zachman, "The information systems management systems: A framework for planning," *Data Base*, Winter 1978.

[42] R. W. Zmud, "Management of large software development efforts," *MIS Quart.*, June 1980.

Dennis S. Kira received the B.Sc. and the M.Sc. degrees in mathematics from Simon Fraser University, and the Ph.D. degree in management science from the University of British Columbia.

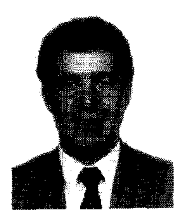
He is Associate Professor in the Department of Decision Sciences and Management Information Systems, Concordia University, Montreal, Canada. His research interests are in DSS, strategic planning models, financial modeling, and the choice models. He has had articles published in *Management Science*, the *Review of Economic Studies*, *Omega*, the *Journal of Environmental Economics and Management*, *Journal of Business*, and *The Decision Making Under Uncertainty*.

Dr. Kira is a member of TIMS, CORS, and DSI.

Martin Kusy was born in Czechoslovakia. He received the B.Comm degree from Sir George Williams University, Montreal, PQ, Canada, the M.B.A. degree from the University of Windsor, Windsor, ON, Canada, and the Ph.D. degree from the University of British Columbia, Vancouver, BC, Canada.

He is presently an Associate Professor of Decision Sciences and M.I.S. and Associate Dean of Graduate Sciences and has served as a Consultant to the Royal Bank of Canada, General Motors of Canada, among others.

His research interests include operations research and operations management. He is presently involved in theoretical and empirical studies in the capital budgeting process and in the development of stochastic integer programming algorithms. He is a member of TIMS and ORSA.



David H. Murray received the bachelor's degree in pure mathematics and the master's degree in business administration, both from Concordia University, Montreal, PQ, Canada.

He is an Information Systems Specialist working for Canadian National Railways in Montreal, Canada. He has a broad background in all aspects of information systems, from both a management and a technical perspective, and is presently responsible for the development of information systems for the marketing department of his firm. Current research interests include the use of information technology for competitive advantage within the transportation sector, and artificial intelligence systems for transportation rate making.

Barbara J. Goranson received the B.A. degree from Carleton University, Ottawa, ON, Canada, and the M.B.A. degree from Concordia University, Montreal, PQ, Canada.

She is currently a Capital Budgeting Specialist at the Royal Bank of Canada in Montreal.